Technical white paper

HP OO 10.x Network Architecture

Table of Contents

Overview

This document addresses the new architecture of HP Operations Orchestration 10.x focusing specifically on the communication between RAS to Central. Included in this document are the motivations behind the selected architecture and the configuration that is required. This document also details the security controls that are included in OO 10.10 that support the new architecture.

At the end of this document you will find two workarounds that can be applied for policy restricted environments.

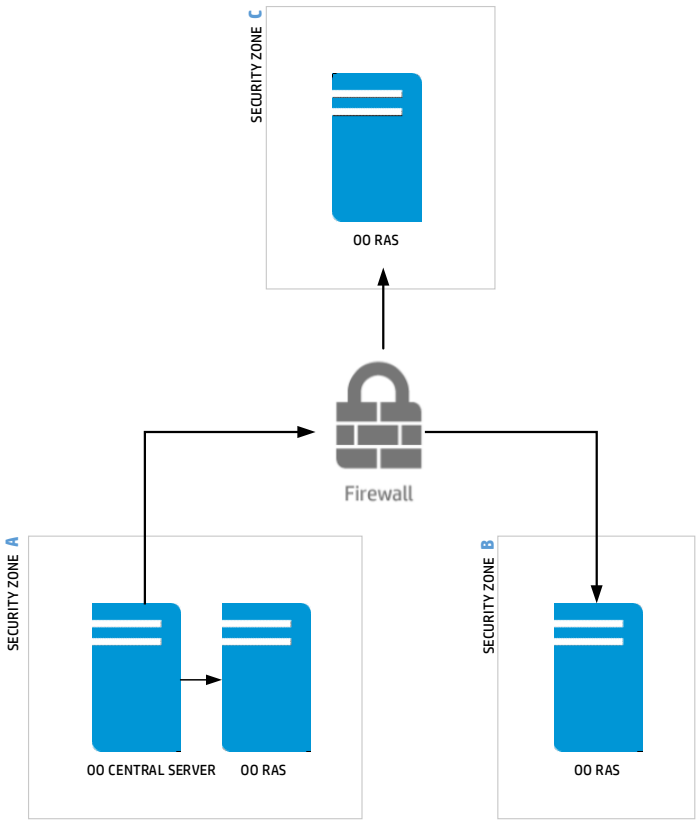
Advancing to a Scalable Model

The Old Model

OO 9.x used a centralized approach for managing all RAS instances connected to the Central cluster. The Central nodes were the sole drivers of the job distribution pipeline. The Central needed to continuously connect to the RAS instances to distribute additional jobs, to query the progress of execution, to query for execution results, detect errors and failures and to identify non-functioning RAS instances.

In this model, all Central\RAS communication were initiated by the Central nodes and therefore all connections were created by the Centrals.

In case RAS instances were deployed cross firewalls, the required firewall setup was to allow connections from Central hosts to RAS hosts, as described in the following diagram:



The main disadvantages to this model are the effects on scalability and performance of the system that are restricted to Central resources. In this case, Central is responsible for all the RASes. The Central has to continue polling all RAS instances for an updated status, push new tasks, and then to query for the results. This utilizes Central resources: threads, CPU cycles, and so on, and poses a physical limit on the amount of RASes a single Central can watch.

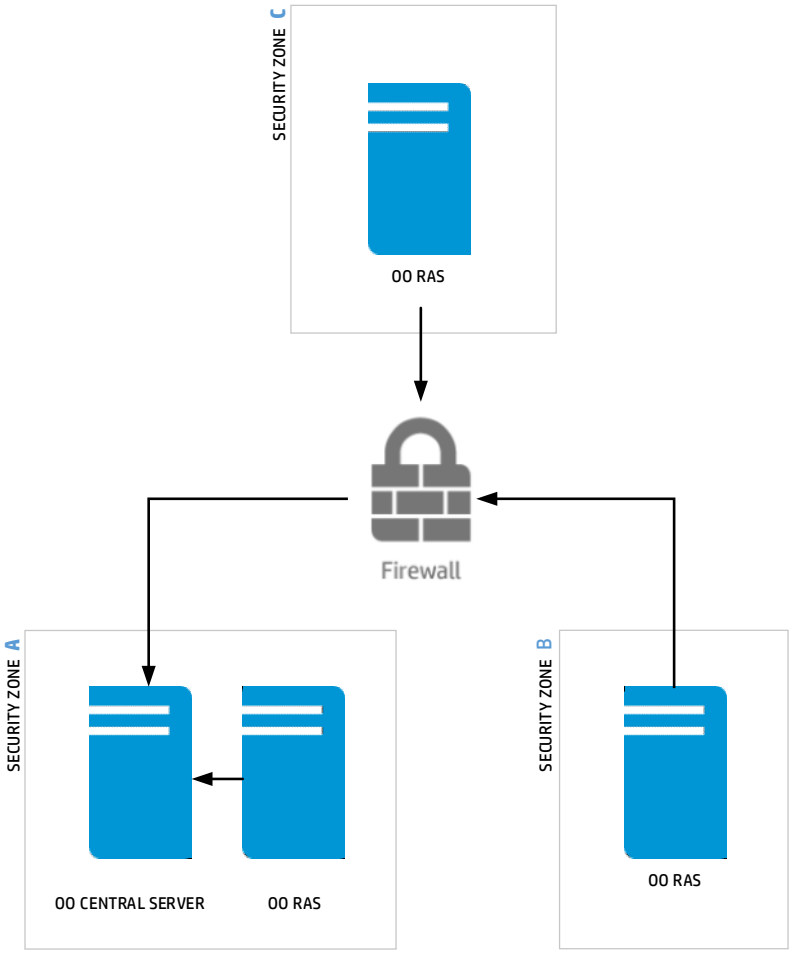
Performance is also impacted. When considering the overall throughput of the system, a single RAS instance cannot maximize its resources since its ability to consume new tasks is limited by the Central availability to dispatch them. Therefore, it is Central resources that are restricting the RAS throughput and not the RAS resources, which is preferred.

The New Model

To overcome this bottleneck, OO 10 introduces a refined architecture model. In the new model, each RAS instance is given the autonomy to manage its own lifecycle. This allows each RAS instance to achieve as high throughput as its own resources allow. It also supports adding more RAS instances to the cluster without exhausting Central resources.

How does it actually work? All RAS\Central communications are now initiated from the RAS. A RAS notifies the Central upon execution status changes, pulls work, and sends a heartbeat to indicate it is alive and well. This guarantees that new work is sent to each worker on the exact timing when it is needed and that RAS is not kept idly waiting for the Central to take the time and notice it is waiting for additional work. It also frees up the Central to manage more RAS instances than the limited amount of system threads available for the Central process.

For this type of communication, all connections are initiated from the RAS towards the Central cluster. In case RAS instances were deployed cross firewalls, the required firewall setup allows connections from RAS hosts to Central hosts (or load balancer), as described in the following diagram:



Configuring the New Model

Firewall Configuration

In case there is a firewall placed between the Central node(s) and a RAS instance, the Central port should be opened in the firewall configuration to allow HTTP(S) requests from the RAS to the Central.

Important
The Central incoming communication port is set by the OO administrator during the Central installation and can be set to any available port on the machine, including 8080 or 8443, as required.

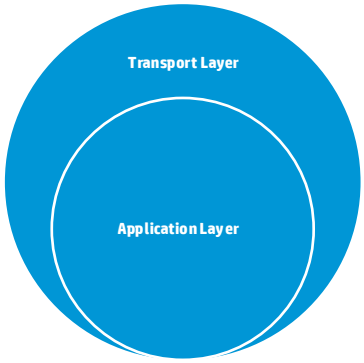
Worker Groups Configuration

Configuring worker groups has become much simpler with the new communication architecture. In OO 9 in order to achieve RAS high availability (HA) and scalability, each of these group of workers (RAS instances) needs to be accessed through a load balancer.

With the new OO 10 communication architecture, each RAS instance approaches the Central directly to pull tasks, and the logical groups are only virtual and managed using the topology module in Central. One of the main advantages to the new model is that a RAS load balancer is no longer needed. For more information on setting up worker groups, see the HP OO 10.x Central Guide.

Security Controls

This section describes the HP OO 10.10 security controls, and how they contribute to the safety of RAS-to-Central communication.
HP OO 10.10 architecture places security controls at multiple layers of the product to maintain security at all times.



Transport Layer

HTTPS

Communication between RAS and Central can be limited to use HTTPS only, therefore protecting confidentiality and integrity of data. The OO administrator is advised to switch the self-signed certificate provided OOTB with a certificate issued by a CA to insure the identity of the RAS.

Client Certificate

To eliminate the risk of RAS identity spoofing, the RAS can be configured to use a client certificate when communicating with the Central.

Push Architecture

RAS has no means of incoming communication. Therefore the managed applications, even those that reside in the same security zone as that specific RAS instance cannot exploit the RAS-to-Central trusted communication channel.

Application Layer

Access Control - Authentication

Simply installing a RAS instance and providing it with the Central URL to communicate with is not sufficient even for the vanilla HTTP communication.

In order for a central node to acknowledge and respond to a RAS instance, a manual approval step performed by the Central user with administrator permissions must be accomplished. The admin user needs to enable the specific RAS using the Central's topology management module.

To verify that Central accepts calls from enabled RAS instances only, every RAS call includes the RAS credentials that are authenticated to make sure that communication is received from an enabled instance.

These RAS credentials are unique for each RAS instance and are granted by the installer upon RAS installation.

Additional Security Controls

Data Volatility

All data that is passed to the RAS is kept in memory for the duration of this specific flow execution only. Once the execution is terminated, or moved to another worker (either in Central or in another RAS) the data is no longer required and will be safely disposed of.

Independent verifications

FIPS 140-2 Level 1

HP OO 10.10 is integrated with third party FIPS 140-2 Level 1 validated cryptographic module RSA BSAFE Crypto-J.

When HP OO is configured to operate in FIPS-compliant mode, the main functions and procedures, for example, SSL/TLS connections, which require cryptography such as secure hash, encryption, digital signature etc. – make use of the crypto services provided by RSA BSAFE Crypto-J. This supports FIPS 140-2 compliance of HP Operations Orchestration.

For more details on how to configure HP OO and its components to conform to FIPS 140-2 Level 1 standard see the HP OO System Configuration and Hardening Guide. Additional references are listed in the Appendix at the end of this document.

Common Criteria

OO is in process of compliancy with Common Criteria.

Applying Workarounds for Policy Restricted Environments

In certain cases, due to policy restrictions, the required firewall configuration cannot be implemented despite the multiple security controls in HP OO 10.10.

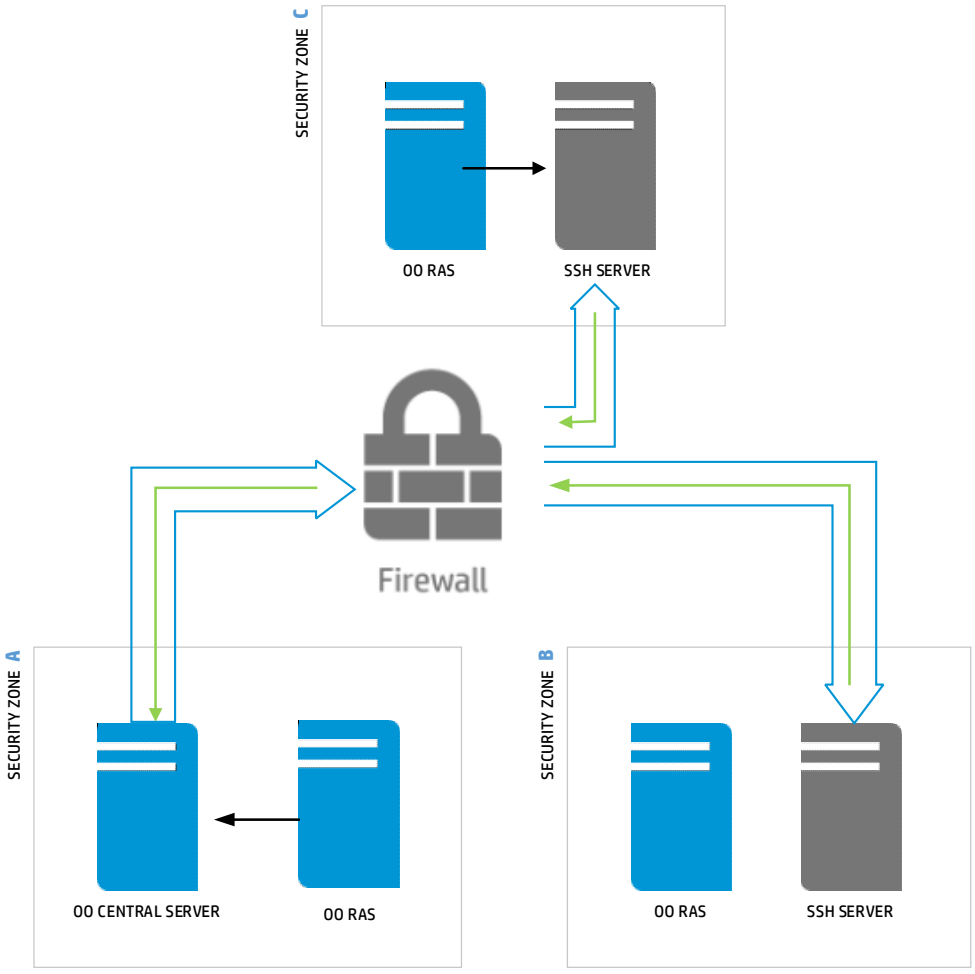
The following workarounds can be applied in these cases to the RAS-to-Central communication channel.

SSH reverse tunneling

In this case, HP OO uses an SSH reverse tunnel that eliminates the need for opening the Central port in the firewall for incoming connections.

Note

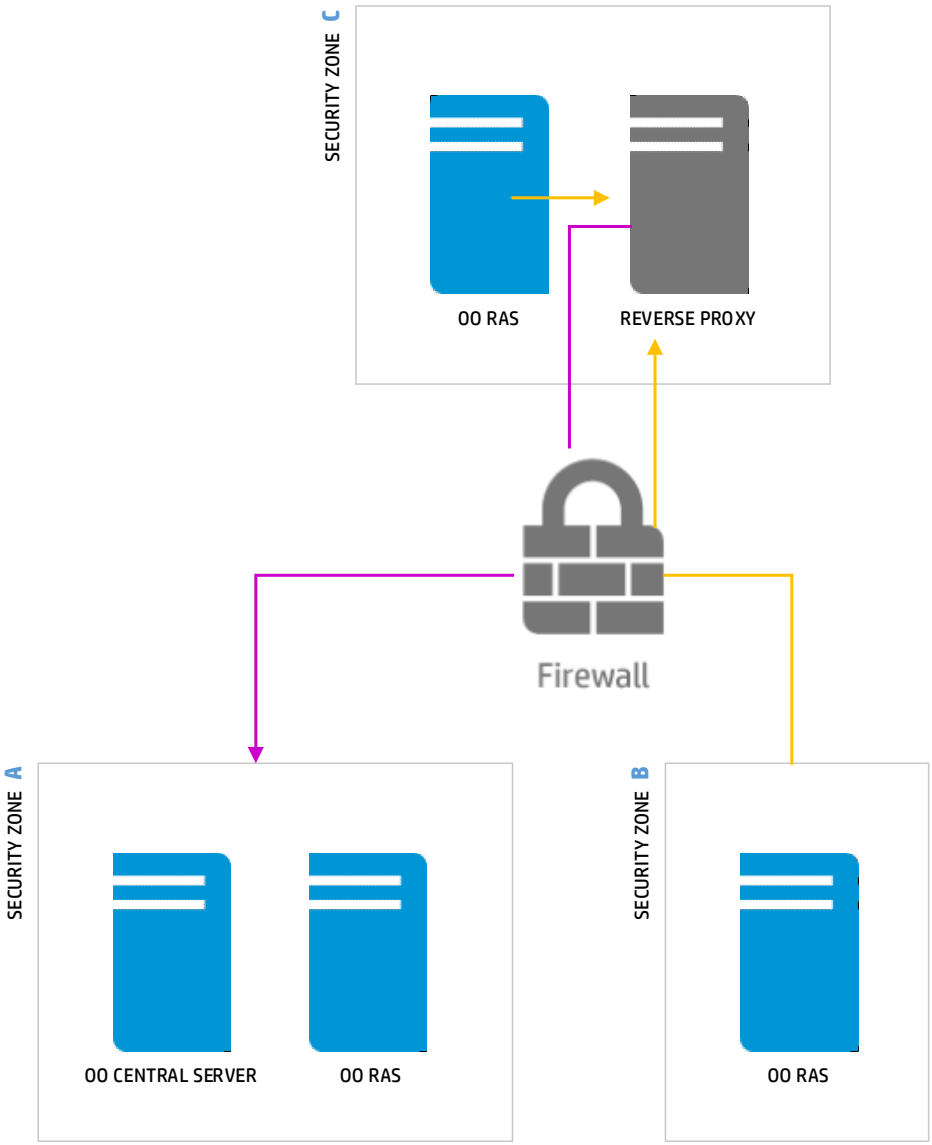
The SSH server can be set up on the same host as the RAS.



See Appendix A for the detailed instructions.

Reverse Proxy

In this workaround, the reverse proxy instance is placed between the RAS instances and the Central nodes. As a result, the Central port can be opened in the firewall to allow connections originated from the reverse proxy only. The reverse proxy instance serves as application-level firewall to provide additional protection on the RAS-to-Central communication channel. To achieve high availability, multiple instances of reverse proxies can be set.



Additional resources

The latest version of the HP OO 10.x documentation is available on the HP OO SSO and HP Live Network site.

Recommended documentation:

- HP OO 10.10 Central Guide
- HP OO10.10 Configuration and Hardening Guide
- HP OO10.10 System Requirements
- HP OO10.10 Architecture Guide

Conclusion

To conclude, OO 10 introduces a new network architecture and multiple security controls to support it. For the specific cases where policies do not allow adoption of the new model as is, several workarounds are offered that you can apply to the model to eliminate the need for the firewall configuration change.

Appendix A

Using SSH reverse tunneling

In order to tunnel RAS-to-Central connection, an SSH connection needs to be established between the Central and RAS servers. We tried several different Central and RAS OS combinations as specified below. These tests used the following SSH client and server implementations:

- For Windows we have used Cygwin for running OPENSSH SSHD server and PuTTY as the SSH client.
  - Cygwin version 1.7
  - PuTTY version 0.63
- For Linux we used the embedded implementations of SSH.

Table 1. Operating systems used:

Windows 2012	RAS
Windows 2008R2	Central
Windows 2012R2	External Network RAS
Red Hat Enterprise Linux 6.4 x64	Linux RAS
Ubuntu 12.04 x64	Central

Note
At the time of writing this document, Cygwin did not support IPv6, which also means that the OpenSSH included with it also does not support IPv6. Therefore, in order to establish connections to it make sure that all ssh connections opened to the server are using IPv4.

Windows Configuration

Setup on RAS

On the RAS server OPENSSH needs to be installed and configured. For this purpose we used Cygwin.

1. Download and install Cygwin x64.
2. During installation of Cygwin make sure to include the following packages:
   a. Cygrunsrv from admin section
   b. Openssh from Net section

   Note
   If you already have Cygwin installed on your server, without these packages, rerun setup to add them to the existing installation.

3. Open a new bash shell window and run the command `ssh-host-config –y`. This creates the necessary configuration files, a privileged user and necessary directories.
4. If prompted with "CYGWIN=" enter: "tty ntsec"
5. When prompted for a password enter a password matching your organization's security standards.
6. Edit the sshd_config file (etc/sshd_config) and change the following:
   - `AllowAgentForwarding yes`
   - `AllowTcpForwarding yes`
   - `GatewayPorts yes`
   - `PermitTunnel yes`

   Note
   In case the lines are commented in the configuration file, make sure to also uncomment the lines when making these changes.

7. Start the service using: `cygrunsrv –S sshd.`

Setup on Central

On the Central server open PuTTY and create a SSH connection to the RAS server. In order to initiate the tunnel, perform the following:

1. With the SSH session open, right-click in the PuTTY window and select **Change settings**.
2. Select **Connections**, expand SSH, and select **tunnels**.
3. In the source port field, enter the port that you want the tunnel to use on the RAS server, for example 9001.
4. In the destination field enter the Central host and Central port that you want to tunnel in the following format:
   `<central_host>:<central_port>`
   For example, `myhost:8080`

Note
It is recommended to use FQDN when entering the Central Host.

---

5. Select **Remote** and IPv4.
6. Click **Add**.
7. Repeat steps 3-6 if you wish to add another port, make sure you use a different source port.
8. When are finished adding forwarded ports, click on **Apply**.

The tunnel is now set up and central will be accessible from the RAS server as long as the SSH connection between Central and RAS is open. When you close the PuTTY window, the tunnel is also closed.

To access Central, enter the following:
http://<ras_host>:<http_source_port>/oo (for example, **http://rashost:9001/oo**), or
https://<ras_host>:<https_source_port>/oo (for example, **https://rashost:9002/oo**), where **http(s)_source_port** is the source port defined in PuTTY for the http(s) destinations.

Linux configuration

Setup on RAS

On the RAS server the SSH server needs to be configured to allow tunneling:

1. Edit **sshd_config**, located in `/etc/ssh/sshd_config` and perform the following changes:
   - `AllowAgentForwarding yes`
   - `AllowTcpForwarding yes`
   - `GatewayPorts yes`
   - `PermitTunnel yes`

   Note
   Some of the lines above may be commented in the **sshd_config** file, uncomment them and make the necessary changes.

   ---

2. Restart ssh:

- CentOS / RHEL / Fedora / Redhat Linux:
  `# /etc/init.d/sshd restart` or `# service sshd restart`
- Debian / Ubuntu Linux
  `# /etc/init.d/ssh restart` or `# service ssh restart`
- FreeBSD
  `# /etc/rc.d/sshd restart`

Setup on Central

On the Central server, open a terminal window and enter the following command:

```
#ssh –R <remote_port>:<central_host>:<central_port> <ras_host>
```

For example, `ssh –R 9001:centralhost:8080 rashost`

- In case you want to forward both http and https ports you will have to open separate terminal windows for each.
- In case you want to use a different user then the user from the Central server use the following command:
  `#ssh –R <remote_port>:<central_host>:<central_port> <user>@<ras_host>`

General Notes

- The tunnel can be created with multiple RASes with any combination of Linux Central – Linux RAS, Linux Central – Windows RAS, Windows Central – Linux RAS, Windows Central – Windows RAS, by using the appropriate configuration for the host.
- The tunnel always has to be initiated from the Central host.
- The user used for tunneling has to have sufficient privileges to run and install HP OO.
- When working with RASes from within the security zone and from outside the security zone make sure you group them accordingly in order to avoid situations where workers attempt to access servers or resources that are not available to them.

More FIPS Information

Compliance was validated according to Cryptographic Module Validation Program

(CMVP http://csrc.nist.gov/groups/STM/cmvp/index.html) and certified by NIST (http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm#1786).